11. Teresa

Program Name: Teresa.java **Input File: teresa.dat**

Teresa's big sister has been learning about matrix operations in her college math class. She doesn't really understand many of the details but has decided to try programming an experimental operation as a personal challenge; however, she is struggling. She is trying to perform a 2-way sort of a matrix so the smallest value is in the top left corner and the largest value is in the bottom right corner. Her experimental operation first sorts within the columns and then across the rows. The result should be that the values increase from left-to-right and top-to-bottom as shown in the following example.

Original Matrix			
73	95	72	15
47	21	41	66
23	61	38	87
98	28	57	39
37	60	22	49

Columns Sorted			
23	21	22	15
37	28	38	39
47	60	41	49
73	61	57	66
98	95	72	87

Final Matrix			
15	21	22	23
28	37	38	39
41	47	49	60
57	61	66	73
72	87	95	98

Can your UIL team help Teresa with this sorting challenge?

Input: First line will contain a number $1 \leq T \leq 10$ as the number of test cases. The first line of each test case will contain two whole numbers R and C, the number of rows and columns in the matrix, both will be ≥ 2 and ≤ 25. The following R rows will each contain exactly C integers, separated by white space, which is the data for the matrix. All data values will be ≥ 100 and < 1000.

Output: For each test case, output one line containing the test case number followed by a colon. The next R rows display the data in the columns of the sorted matrix as shown below, with exactly one tab following each data value. Below each test case, display a single line containing 12 plus signs, "++++++++++++"

Sample input:

```
2
5 4
194    819    449    405
560    410    914    534
302    670    856    448
933    239    259    477
591    455    665    652
3 5
558    777    773    761    995
252    632    580    639    818
590    828    372    386    489
```

Sample output:

```
1:
194    239    259    405
302    410    448    449
455    477    560    665
534    591    670    856
652    819    914    933
++++++++++++
2:
252    372    386    489    632
558    580    639    777    818
590    761    773    828    995
++++++++++++
```